

# Lecture 2: Monoalphabetic Ciphers

Linguistics 484  
Winter 2008

# Cryptosystems

- ✱ A cryptosystem is an arrangement that allows A to send secure messages to B
- ✱ What is a message?

# What is a message?

- \* A message is a sequence of symbols, where each symbol is drawn from a defined alphabet.
- \* For example, in many codes, the alphabet is A-Z plus SPACE

# Alternate alphabets

- ✱ Bacon's alphabet has only 24 different symbols, because I/J and U/V use the same encodings.
- ✱ The standard Enigma alphabet uses X to indicate spaces (or sometimes just leaves out the spaces)

# Using alphabets

- ✱ To write in Bacon's alphabet, you just do the same thing for I as you do for J
- ✱ To read the result, you have to look at context and guess which it is.
- ✱ The message written in the 24 character alphabet has less information than the original. But the reader restores it.
- ✱ For Enigma, an X is usually a space, rarely itself.

# Redundancy

- ✱ The reason why it is usually possible to restore the original is that the message is redundant.
- ✱ Redundancy is a useful property of human language and many other communication systems.

# Redundancy

- ✱ Informal definition of redundancy. If a message is redundant, then you can use one part of the message to help you guess another part.
- ✱ There is math that goes with this. It is called Information Theory

# Formal languages

# Formal languages

- ✱ A **language** (L) is a set of possible messages.

# Formal languages

- \* A **language** (L) is a set of possible messages.
- \* It has an **alphabet** (V) which is the set of all the possible characters in the messages.  
e.g.  $V = \{A-Z + \text{space}\}$

# Formal languages

- \* A **language** (L) is a set of possible messages.
- \* It has an **alphabet** (V) which is the set of all the possible characters in the messages.  
e.g.  $V = \{A-Z + \text{space}\}$
- \* The alphabet defines all the possible single character messages.

# Formal languages

- \* A **language** (L) is a set of possible messages.
- \* It has an **alphabet** (V) which is the set of all the possible characters in the messages.  
e.g.  $V = \text{/A-Z + space/}$
- \* The alphabet defines all the possible single character messages.
- \* The set of possible messages is notated as  $L = V^*$  where  $V^*$  means “zero or more of V”

# Cryptosystems

- \* A cryptosystem relates two formal languages:  
 $L_{\text{Plain}}, L_{\text{Cipher}}$
- \*  $L_{\text{Plain}}$  is the language of the plaintext,  $L_{\text{Cipher}}$  that of the ciphertext, with two alphabets:  $V_{\text{Plain}}, V_{\text{Cipher}}$



# Properties of cryptosystems

- \* Many cryptosystems have the same alphabet for both sides of the mapping. If so, they are called **endomorphic**.
- \* Some cryptosystems preserve message length. The ciphertext is guaranteed to be the same length as the original plaintext.
- \* Some cryptosystems guarantee that no letter in the plaintext will ever be mapped to the same letter in the cryptotext.

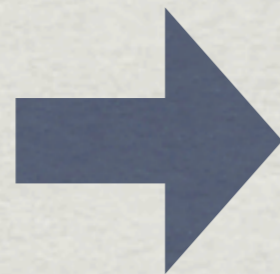
# ROT13

- \* ROT13 is the name for a shift cipher that shifts letters 13 places forward in the alphabet.
- \* This has been used a lot to post puzzles and jokes in emails without giving away the punchline or accidentally exposing people to rude words.

# ROT13

Plain	A	B	C	D	E	F	G	H	I	J	K	L	M
Cipher	n	o	p	q	r	s	t	u	v	w	x	y	z
Plain	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Cipher	a	b	c	d	e	f	g	h	i	j	k	l	m

**BOOK**



**OBBX**

# Why ROT13?

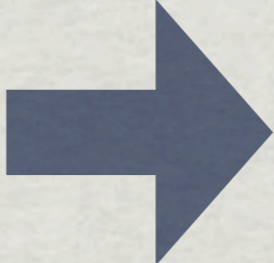
Plain	A	B	C	D	E	F	G	H	I	J	K	L	M
Cipher	n	o	p	q	r	s	t	u	v	w	x	y	z
Plain	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Cipher	a	b	c	d	e	f	g	h	i	j	k	l	m

**BOOK**

**OBBX**

# Why ROT13?

Plain	A	B	C	D	E	F	G	H	I	J	K	L	M
Cipher	n	o	p	q	r	s	t	u	v	w	x	y	z
Plain	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Cipher	a	b	c	d	e	f	g	h	i	j	k	l	m

**BOOK**  **OBBX**

# Why ROT13?

Plain	A	B	C	D	E	F	G	H	I	J	K	L	M
Cipher	n	o	p	q	r	s	t	u	v	w	x	y	z
Plain	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Cipher	a	b	c	d	e	f	g	h	i	j	k	l	m

**BOOK**

**OBBX**

# Why ROT13?

Plain	A	B	C	D	E	F	G	H	I	J	K	L	M
Cipher	n	o	p	q	r	s	t	u	v	w	x	y	z
Plain	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Cipher	a	b	c	d	e	f	g	h	i	j	k	l	m



# Why ROT13?

Plain	A	B	C	D	E	F	G	H	I	J	K	L	M
Cipher	n	o	p	q	r	s	t	u	v	w	x	y	z
Plain	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Cipher	a	b	c	d	e	f	g	h	i	j	k	l	m



**ROT13 IS A RECIPROCAL CIPHER**

# furrfu

ABBA -> NOON

ABJURER -> NOWHERE

AH -> NU

ANT -> NAG

BALK -> ONYX

BAR -> ONE

BE -> OR

CLERK -> PYREX

CRAG -> PENT

EBBS -> ROOF

ENG -> RAT

ENVY -> RAIL

ERRS -> REEF

FABER -> SNORE

FUR -> SHE

GEL -> TRY

GNAT -> TANG

GREEN -> TERRA

INK -> VAX

IRK -> VEX

JUNES -> WHARF

JURA -> WHEN

# furrfu = sheesh

ABBA -> NOON

ABJURER -> NOWHERE

AH -> NU

ANT -> NAG

BALK -> ONYX

BAR -> ONE

BE -> OR

CLERK -> PYREX

CRAG -> PENT

EBBS -> ROOF

ENG -> RAT

ENVY -> RAIL

ERRS -> REEF

FABER -> SNORE

FUR -> SHE

GEL -> TRY

GNAT -> TANG

GREEN -> TERRA

INK -> VAX

IRK -> VEX

JUNES -> WHARF

JURA -> WHEN

# ROT13 is not secure!

- ✱ But that didn't stop Netscape from using it in a password "protection" mechanism.
- ✱ All it is good for is helping people not to read more than they want.

# Reciprocal ciphers

- \* Being reciprocal is another mathematical property of that endomorphic cryptosystems can have.
- \* Reciprocal ciphers are convenient, because you use the same settings and program for encryption and decryption. Removes a possible source of error.
- \* But may introduce vulnerabilities, making the cipher easier to break.

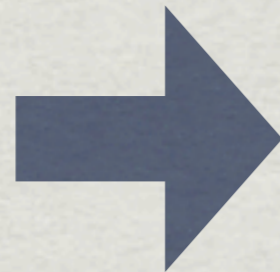
# Shift ciphers

- \* Generalization of ROT13, to ROT1, ROT2, ROT3,...
- \* No longer reciprocal. Still endomorphic.

# ROT1

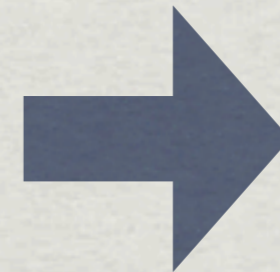
Plain	A	B	C	D	E	F	G	H	I	J	K	L	M
Cipher	b	c	d	e	f	g	h	i	j	k	l	m	n
Plain	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Cipher	o	p	q	r	s	t	u	v	w	x	y	z	a

**BOOK**



**CPPL**

**CPPL**

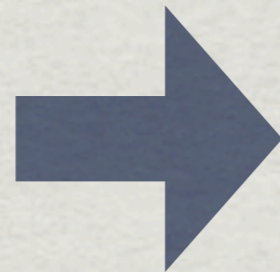


**DQQM**

# ROT25

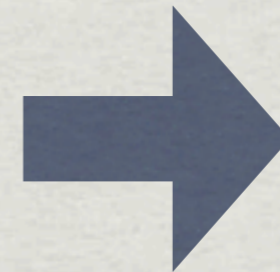
Plain	A	B	C	D	E	F	G	H	I	J	K	L	M
Cipher	z	a	b	c	d	e	f	g	h	i	j	k	l
Plain	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Cipher	m	n	o	p	q	r	s	t	u	v	w	x	y

**DQQM**



**CPPL**

**CPPL**



**BOOK**

# Inverses

- ✱ Rot25 is the **inverse** of Rot1.
- ✱ Rot13 is its own inverse.
- ✱ What is the inverse of Rot7?

# Conan-Doyle's Dancing Man Cipher



[Read: http://en.wikisource.org/wiki/  
The Adventure of the Dancing Men](http://en.wikisource.org/wiki/The_Adventure_of_the_Dancing_Men)

# How hard?

# How hard?

- ✱ Is there a real difference between an endomorphic system and one that uses funny symbols?

# How hard?

- ✱ Is there a real difference between an endomorphic system and one that uses funny symbols?
- ✱ No. Because the relationship between the ciphertext and the plaintext is just the same.

# How hard?

- ✱ Is there a real difference between an endomorphic system and one that uses funny symbols?
- ✱ No. Because the relationship between the ciphertext and the plaintext is just the same.
- ✱ Does it get harder if you do two shift ciphers, one after another?

# How hard?

- ✱ Is there a real difference between an endomorphic system and one that uses funny symbols?
- ✱ No. Because the relationship between the ciphertext and the plaintext is just the same.
- ✱ Does it get harder if you do two shift ciphers, one after another?
- ✱ No. Because doing two (or more) is always exactly equivalent to one of the 26 possible shift ciphers.

# Spurious complexity

- \* Using funny symbols doesn't make the code any harder.
- \* Because you could systematically change all the symbols to e.g. 1,2,3,... and all the information would still be there.
- \* When you design a code, using funny symbols annoys everyone, but the complexity that it introduces is spurious.

Is the dancing man a  
shift cipher?

# Is the dancing man a shift cipher?

- \* No. Because it isn't endomorphic

# Is the dancing man a shift cipher?

- \* No. Because it isn't endomorphic
- \* No. Because there is no natural order for weird dancing figures.

# Cryptanalysis of shift ciphers

- \* There are only 26 possible shift ciphers
- \* This means that by exploring 26 possibilities, we will find all the possible decryptions.
- \* How can we do this systematically?

**CPZPAPUNHBUAZJHUILHWYVISLT**

# OK. Now what?

- ✱ In this case, only one line looks plausible, so we have an answer.
- ✱ What happens if you do the same to DTCFAECTA?

D	T	C	F	A	E	C	T	A
E	U	D						
F	V	E						
G	W	F						
H	X	G						
I	Y	H						
J	Z	I						
K	A	J						
L	B	K						
M	C	L						
N	D	M						
O	E	N						
P	F	O						
Q	G	P						
R	H	Q						
<b>S</b>	<b>I</b>	<b>R</b>	<b>U</b>	<b>P</b>	<b>T</b>	<b>R</b>	<b>I</b>	<b>P</b>
T	J	R						
U	K	S						
V	L	T						
W	M	U						
X	N	V						
Y	O	W						
Z	P	X						
A	Q	Y						
<b>B</b>	<b>R</b>	<b>A</b>	<b>D</b>	<b>Y</b>	<b>C</b>	<b>A</b>	<b>R</b>	<b>Y</b>
C	S	B	E	Z	D	B	S	Z

# Unicity distance

- \* Obviously, a one character message is too short to decode unambiguously even if you know it is a shift cipher.
- \* But you need only a few characters, because the range of possible keys for shift ciphers is just the numbers 1-25, then you are almost always sure.
- \* Shift ciphers have a small **unicity distance**.  
Ciphers with more elaborate keys need more text.

# Eve, Trent, Mallory

- \* Eve: an eavesdropper (usually just listens)
- \* Trent: a trusted person for both Alice and Bob.
- \* Mallory: a malefactor who might change the message as it passes from Alice to Bob.

# What is security?

- \* The way to analyse security is to think about specific attacks.
- \* We can't ever just say a message is secure.
- \* Personifying the participants helps make discussion concrete.

# Alice, Bob

- ✱ It's boring to talk about  $A$  and  $B$ , so the crypto world personifies them as Alice, who delivers a message to Bob